Attorney Docket No.: 42390P10683
Express Mail No.: EL851136790US

UNITED STATES PATENT APPLICATION

FOR

SYSTEM AND METHOD TO IMPLEMENT AN IMPROVED FRAME FRAGMENTATION PROTOCOL TO FACILITATE EFFICIENT TRANSMISSION OF PRIORITIZED DATA

Inventor:

Bent S. Jensen

Prepared By:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
12400 Wilshire Blvd., 7th Floor
Los Angeles, California 90025-1026
(714) 557-3800

BACKGROUND

(1) Field

The present invention relates to a system and method to implement an improved frame fragmentation protocol to facilitate efficient transmission of prioritized data.

(2) General Background

Currently, there are a number of techniques to process data streams in which high priority data and low priority data are mixed. Examples of these techniques include Quality of Service Point-to-Point Protocol (QoSPPP) and a processing technique proposed in Request for Comments 2687 (RFC 2687), "PPP in a Real-time Oriented HDLC-like Framing", published in September 1999.

The aim of QoSPPP is to allow a transmitting unit to run a mix of applications with varying communications needs. Currently most Point-to-Point Protocol (PPP) implementations offer a single class of service, best-effort, which is most suited for conventional data applications (e.g., Telnet, ftp, email, etc.). However, newer Internet applications such as packet telephony, video conferencing, etc., require a new class of service with bandwidth guarantees and upper bounds of the delay and jitter seen by their packets. QoSPPP supports four classes of service, namely Available Bit Rate (ABR), Unspecified Bit Rate (UBR), Constant Bit Rate (CBR), and Variable Bit Rate (VBR).

ABR supports traditional data applications, which do not need bandwidth guarantees or any strict bounds on delays and jitters. These data applications typically have variable sized packets. However, ABR applications will specify their maximum datagram size, expected bandwidth usage, and maximum tolerable delays. The class of service is specified the flowspec along with other

parameters like bandwidth, delay, and jitter. While the network does not guarantee delays and jitters, it uses them to estimate buffer sizes and expected load. UBR or Unspecified Bit Rate is for legacy applications that are not aware of the Quality of Service (QoS).

CBR or Constant Bit Rate is for applications that transmit data at regular intervals. Datagrams are usually small and has fixed length (though the latter is not a requirement). An example is a packet phone that does not perform silence detection. Datagrams have strict upper bounds on delay and jitter that can be tolerated and also on bandwidth requirements. VBR or Variable Bit Rate is similar to CBR, except that the rate of packet transmission is not fixed.

RFC 2687 generally proposes suspend/resume mechanism and also multiple classes to obtain multiple levels of suspension. However, the applicability of the multilink header for a suspend/resume mechanism is limited, as the "end" bit is in the multilink header, which is the wrong location for suspend/resume operation. To suspend a big packet, the packet must be sent with the "end" bit off, and (unless the packet was suspended a small number of bytes before its end) and an empty fragment has to be sent afterwards to "close" the packet. The minimum overhead for suspending a packet thus is twice the multilink header size (six bytes, including a compressed multilink protocol field) plus one PPP framing (three bytes). Each suspension costs another six bytes (not counting the overhead of the framing for the intervening packet).

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is an exemplary diagram of a system in accordance with one embodiment of the present invention;

Figure 2 is an exemplary block diagram of a sending unit and a receiving unit in accordance with one embodiment of the present invention;

Figures 3 and 3A illustrate the format of an exemplary frame fragment in accordance with one embodiment of the present invention;

Figure 4 shows an example of frame fragments generated from a given frame in accordance with one embodiment of the present invention;

Figure 5 illustrates an example where transmission of frames having high priority data is promoted over frames having low priority data in accordance with one embodiment of the present invention;

Figure 6 illustrates an exemplary order in which the receiving unit assembles frames after receiving frame fragments in accordance with one embodiment of the present invention;

Figure 7 generally outlines an exemplary process of promoting the transmission of high priority frames over the transmission of low priority frames using the frame fragmentation technique in accordance with one embodiment of the present invention; and

Figure 8 generally outlines an exemplary process of generating frame fragmentation control information in accordance with one embodiment of the present invention.

DETAILED DESCRIPTION

The present invention relates to a system and method to implement an improved frame fragmentation protocol to facilitate efficient transmission of prioritized data.

Figure 1 is an exemplary diagram of a system 100 in accordance with one embodiment of the present invention. The system 100 includes network switches or routers 102, 104 that are operatively coupled together by network links 106,108 and network 110.

Network switch or router 102 is coupled to a plurality of network devices 112, 114, 116, 118, 120. Network devices are generally computing devices having networking capability. As illustrated in Figure 1, examples of network devices can include a laptop computer 112, a desktop computer 114, a network printer 116, a network storage device 118, and a server 120. In practice, a network device can be a set-top-box, a hand-held device, or any computing devices with networking capability.

Network switch or router 104 is coupled to a plurality of network devices, including a server 122, a network storage device 124, a network printer 126, and a desktop 128. Network switch or router is also coupled to a private branch exchange (PBX) system 130. PBX system 130 is coupled to telephones 132, 134 and fax machine 136.

Each device in the system 100 can be a sending unit, a receiving unit, or both. A sending unit is generally a device that transmits data to a receiving unit. A receiving unit is generally a device that receives data transmitted by the sending unit.

Figure 2 is an exemplary block diagram of a sending unit 205 and a receiving unit 210 in accordance with one embodiment of the present invention. Sending unit 205 can include a channel segregator 215, a frame fragment generator 220, and a data transmitter 225. Channel segregator 215 receives input data frames 230, which can come from sources internal or external to the sending unit 205 and which is designated for a logical communication channel 235_1, 235_2, ..., 235_N where "N" is a positive integer. Channel segregator 215 generally places input data frames 230 on the designated logical communication channel 235_1, 235_2, ..., 235_N to forward the input data frames 230 to the frame fragment generator 220.

Frame fragment generator 220 receives and processes input data frames 230 forwarded by channel segregator 215. When necessary, frame fragment generator 220 breaks the input data frames 230 into data segments, generates frame fragments to encapsulate these data segments, and forwards these frame fragments to data transmitter 225. Data transmitter 225 sends the frame fragments to receiving unit 210 via communication link 240. Additional details about the format of frame fragments will be provided below in Figure 3 and the description of the figure.

In general, sending unit 205 generates frame fragments to promote transmission of frames having high priority data over frames having low priority data. An example, where the sending unit generates frame fragments to promote transmission of higher priority frames over low priority data, will be provided below in Figures 5 and 6 and the text describing these figures.

Receiving unit 210 can include a data receiver 245, a frame fragment assembler 250, and a channel aggregator 255. Data receiver 245 extracts frame fragments from communication link 240 and forwards the frame fragments to frame fragment assembler 250. Frame fragment assembler 250 combines the frame fragments into frames and forwards the frames to channel aggregator 255 through

a designated logical communication channel 260_1, 260_2, …, 260_N. Channel aggregator 255 extracts data frames from logical communication channels and forwards or outputs the data frames to destinations internal or external to the receiving unit 210.

It should also be noted that the functional components, as shown in Figure 2 and described in the text accompanying the figure, could be implemented in hardware. However, these functional components can also be implemented using software code segments. Each of the code segments may include one or more programming instructions. If the aforementioned functional components are implemented using software code segments, these code segments can be stored on a machine-readable medium, such as floppy disk, hard drive, CD-ROM, DVD, tape, memory, or any storage device that is accessible by a computing machine.

Figure 3 illustrates the format of an exemplary frame fragment 300 in accordance with one embodiment of the present invention. The exemplary frame fragment 300 can include payload data 305 and frame fragmentation control information 310. Payload data 305 is generally a data segment extracted from a data frame. Frame fragmentation control information 310 generally includes information that can be used to assemble frame fragments 300 into frames.

As shown in the figure, frame fragmentation control information 310 is located at the end of a frame fragment 300. The strategic placement of frame fragmentation control information 310 at the end of a frame fragment 300 provides certain advantages in promoting the transmission of high priority frames over the transmission of low priority frames, as will be shown below in Figure 5 and the description of the figure.

In one embodiment, frame fragmentation control information 310 can include the following fields: first frame fragment indicator (FFFI) 315, frame

fragment sequence number (FFSN) 320, channel number (CN) 325, and last frame fragment indicator (LFFI) 330. First frame fragment indicator 315 specifies whether a frame fragment is a first fragment generated from a frame. Frame fragment sequence number 320 specifies a sequential order number assigned to each frame fragment generated from a frame. Channel number 325 indicates the logical communication channel to which the frame fragment is designated. Channel number 325 is generally used to differentiate multiple payload data flows. Last frame fragment indicator 330 specifies whether a frame fragment is a last fragment generated from a frame.

Frame fragment control information 310 can also include an optional extension indicator 335. The extension indicator 335 is generally used to extend or add fields to the frame fragment control information 310. If the frame fragmentation control information 310 only includes EI 335, FFFI 315, FFSN 320, CN 325, and LFFI 330, extension indicator 335 would be set to FALSE. On the other hand, if frame fragmentation control information includes field or fields in addition to EI 335, FFFI 315, FFSN 320, CN 325, and LFFI 330, extension indicator 335 would be set to TRUE. In one embodiment, FALSE can be represented by a value of one (1), and TRUE can be represented by a value of zero (0).

Figure 3A shows one exemplary implementation of the frame fragment shown in Figure 3 in accordance with one embodiment of the present invention. In this implementation, frame fragment control information 310 occupies the last octet of the frame fragment 300. In the last octet of the frame fragment, FFFI 315 is a one-bit field and occupies the first bit of the octet; FFSN 320 is a three-bit field and occupies the second, third, and fourth bits of the octet; CN 325 is a two-bit field and occupies the fifth and sixth bits of the octet; LFFI 330 is a one-bit field and occupies the seventh bit of the octet; and EI 335 is a one-bit field and occupies the eighth or last bit of the octet.

Figure 4 shows an example of frame fragments 405_1, 405_2, 405_3 generated from a given frame 400 in accordance with one embodiment of the present invention. In the example, the given frame 400 is designated for logical communication channel X, where X is a positive integer. As shown in the figure, the frame is broken up into three data segments 410_1, 410_2, 410_3. Three frame fragments 405_1, 405_2, 405_3 are generated to encapsulate the three data segments 410_1, 410_2, 410_3. The first frame fragment 405_1 includes the first data segment 410_1 of the frame 400, the second frame fragment 405_2 includes the second data segment 410_2 of the frame 400, and the third frame fragment 405_3 includes the third data segment 410_3 of the frame 400.

Each frame fragment 405_1, 405_2, 405_3 includes a frame fragmentation control information 415_1, 415_2, 415_3. The channel number fields 420_1, 420_2, 420_3 in the frame fragmentation control information 415_1, 415_2, 415_3 of each frame fragment 405_1, 405_2, 405_3 are set to X to indicate that the frame fragments 405_1, 405_2, 405_3 are generated from a frame 400 designated for logical communication channel X. Moreover, the extension indicator fields 425_1, 425_2, 425_3 in the frame fragmentation control information 415_1, 415_2, 415_3 of each frame fragment 405_1, 405_2, 405_3 are set to FALSE to specify that there is no extended field.

In frame fragment 405_1, FFFI 430_1 is set to TRUE and FFSN 440_1 is set to 0, indicating that the fragment 405_1 is the first frame fragment generated from the given frame 400. Accordingly, LFFI 435_1 is set to FALSE since frame fragment 405_1 is not the last fragment generated from the given frame 400.

Furthermore, FFFI 430_2 and LFFI 435_2 fields of frame fragment 405_2 are set to FALSE since the fragment 405_2 is neither the first frame fragment nor the last

frame fragment generated from the given frame 400. FFSN 440_2 is set to 1, indicating that the fragment 405_2 is the second frame fragment generated from the given frame 400.

In addition, LFFI 435_3 of frame fragment 405_3 is set to TRUE, indicating that the fragment 405_3 is the last frame fragment generated from the given frame 400. Accordingly, FFFI 430_3 is set to FALSE since frame fragment 405_3 is not the first frame fragment generated from the given frame 400. FFSN 440_3 is set to 2, specifying that the fragment 405_3 is the third frame fragment generated from the given frame 400.

As stated above, a sending unit 205 (shown in Figure 2) generates frame segments from a given frame to generally promote transmission of frames having high priority data over frames having low priority data. Figure 5 illustrates an example where transmission of frames having high priority data is promoted over frames having low priority data in accordance with one embodiment of the present invention. The figure shows a time line having points in time (t1, t2, t3, t4, t5, t6, t7, t8, t9, t10, and t11) when pertinent events occur.

A sending unit begins to receive frame A at time t1 and starts to transmit frame A 505 at time t2. Prior to the complete reception of frame A 505, the sending unit starts to receive frame B 510 at time t3. Frame B 510 has higher priority than Frame A 505. As a result, the sending unit promotes transmission of frame B 510 over the transmission of frame A 505. To do so, the sending unit packages the first data segment 520_1 of frame A 505 in frame fragment A1 525_1. The first data segment 520_1 of frame A 505 generally includes the portion of frame A that the sending unit receives from time t1 to time t3. The sending unit also includes frame fragmentation control information 530_1 in frame fragment A1 525_1.

As previously stated, frame fragmentation control information 530_1, 530_2, 530_3, 530_4 is generally used in the process of assembling frame fragments 525_1, 525_2, 525_3, 525_4 into frames.

At time t4, the sending unit completes the transmission of frame fragment A1 525_1. The sending unit then packages frame B 510 in frame fragment B 525_2 and includes frame fragmentation control information 530_2 in frame fragment B 525_2. The sending unit begins transmission of frame fragment B 525_2 at time t5. At time t7, the sending unit completes the transmission of frame fragment B 525_2.

Prior to the completion of the transmission of frame fragment B 525_2, the sending unit begins to receive frame C 515 at time t6. Since frame B 510 and frame C 515 have similar priority (i.e., high priority), the sending unit does not promote the transmission of frame C 515 over the transmission of frame B 510. However since frame C 515 has higher priority than frame A 505, the sending unit promotes the transmission of frame C 515 over the transmission of the second data segment 520_2 of frame A 505. As such, the transmission of frame fragment C 525_3 begins after the completion of the transmission of frame fragment B 525_2 and before the start of the transmission of frame fragment A2 525_4.

The sending unit packages frame C 515 in frame fragment C 525_3 and includes frame fragmentation control information 530_3 in frame segment C 525_3. At time t8, the sending unit begins transmitting frame fragment C 525_3. The sending unit completes the transmission of frame fragment C 525_3 at time t9.

After the transmission of frame segment C 525_3 is completed, the sending unit resumes the transmission of the second data segment 520_2 of frame A 505. The sending unit packages the second data segment 520_2 of frame A 505 in frame

fragment A2 525_4. The sending unit includes frame fragmentation control information 530_4 in frame fragment A2 525_4. At time t10, the sending unit begins transmitting frame fragment A2 525_4. At time t11, the sending unit completes the transmission of frame fragment A2 525_4.

It should be noted that including frame fragmentation control information at the end of frame fragments enables promotion of transmission of frames having high priority data over frames having low priority data. In the midst of transmitting one frame, the sending unit may decide to suspend the transmission to promote transmission of another frame having higher priority than the current frame (i.e., the frame currently being transmitted). To suspend the transmission of the current frame, the sending unit merely needs to add frame fragmentation control information to the portion of the current frame that has already been transmitted. Afterward, the sending unit can begin transmitting the frame with high priority data. Figure 5, as shown and described above, illustrates an example of a scenario in which transmission of a low priority frame is suspended to promote transmission of high priority frames.

Figure 6 illustrates an order in which the receiving unit 210 (shown in Figure 2) assembles frames 605, 610, 615 after receiving frame fragments 525_1, 525_2, 525_3, 525_4 (shown in Figure 5) in accordance with one embodiment of the present invention. The figure shows a time line having points in time (s1, s2, s3, s4, s5, and s6) when pertinent events occur. As shown in Figure 5, frame fragments 525_1, 525_2, 525_3, 525_4 are transmitted in the following sequential order: frame fragment A1 525_1, frame fragment B 525_2, frame fragment C 525_3, and frame fragment A2 525_4.

Returning to Figure 6, the receiving unit reassembles the frame fragments 525_1, 525_2, 525_3, 525_4 to generate frames 605, 610, 615. The frames are generated in

the following sequential order: frame B 605, frame C 610, and frame A 615. As shown in the figure, receiving unit starts generating frame B 605 at time s1 and completes generating the frame 605 at time s2. Subsequently, receiving unit begins generating frame C 610 at time s3 and finishes generating frame C 610 at time s4. Afterward, the receiving unit starts the process of assembling and generation of frame A 615 at time s5 and completes the process at time s6. Frame A 615 is assembled and generated after frames B 605 and C 610 are generated since frame fragment A2 525_4 was transmitted after the transmission of frame fragments B 525_2 and C 525_3, as shown in Figure 5.

Figure 7 generally outlines the process 700 of promoting the transmission of high priority frames over the transmission of low priority frames using the frame fragmentation technique in accordance with one embodiment of the present invention. In block 705, the first frame fragment is transmitted. The first frame fragment includes the first data segment extracted from a low priority frame and frame fragmentation control information appended to the end of the first data segment. An example of the first frame fragment is shown in Figure 4. The process of generating frame fragmentation control information is outlined below in Figure 8 and the description of the figure.

In block 710, a second frame fragment is transmitted. The second frame fragment includes a high priority frame. The second frame fragment further includes frame fragmentation control information appended to the end of the high priority frame. An example of the second frame fragment, including a high priority frame and frame fragmentation control information appended to the end of the high priority frame, is shown in Figure 4. The process of generating frame fragmentation control information is outlined below in Figure 8 and the description of the figure.

In block 715, a third frame fragment is transmitted. The third frame fragment includes the second data segment extracted from the low priority frame. The third frame fragment further includes frame fragmentation control information appended to the end of the second data segment. An example of a frame fragment, including the second data segment extracted from the low priority frame and frame fragmentation control information appended to the second data segment, is illustrated in Figure 4. The process of generating frame fragmentation control information is outlined below in Figure 8 and the description of the figure.

Figure 8 generally outlines the process 800 of generating frame fragmentation control information in accordance with one embodiment of the present invention. An exemplary format of the frame fragmentation control information 310 is shown in Figure 3. As shown in the figure, in one embodiment, frame fragmentation control information 310 can include the following fields: first frame fragment indicator 315, frame fragment sequence number 320, channel number 325, and last frame fragment indicator 330.

Returning to Figure 8, the first frame fragment indicator is inserted in the frame fragmentation control information in block 805. As stated above, first frame fragment indicator specifies whether a frame fragment is a first fragment generated from a frame. In block 810, frame fragment sequence number is inserted in the frame fragmentation control information to generally specify a sequential order number assigned to the frame fragment.

Channel number is inserted the frame fragmentation control information in block 815. As previously stated, channel number indicates the logical communication channel to which the frame fragment is designated. Channel number is generally used to differentiate multiple payload data flows. In block 820, last frame fragment indicator is inserted in the frame fragmentation control

information to specify whether the frame fragment is the last fragment generated from a frame.

The extension indicator is inserted in the frame fragment control information in block 825. As stated above, the extension indicator is generally used to extend or add fields to the frame fragment control information.

While certain exemplary embodiments have been described and shown in accompanying drawings, it is to be understood that such embodiments are merely illustrative of and not restrictive on the broad invention, and that this invention not be limited to the specific constructions and arrangements shown and described, since various other modifications may occur to those ordinarily skilled in the art.